(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
21 June 2001 (21.06.2001)

PCT

(10) International Publication Number
WO 01/44992 A1

(51) International Patent Classification[7]: G06F 17/30

(21) International Application Number: PCT/US00/41713

(22) International Filing Date: 31 October 2000 (31.10.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/170,974 15 December 1999 (15.12.1999) US

(71) Applicant: **YELLOWBRIX, INC.** [US/US]; Suite 700, 66 Canal Center Plaza, Alexandria, VA 22314 (US).

(72) Inventors: **WIESER, Jace, D.**; 10208 Cedar Pond Dr., Vienna, VA 22182 (US). **AMIN, Anjal, P.**; Unit 1, 5914 Coverdale Way, Alexandria, VA 22310 (US). **MASSA, Jeffrey, P.**; 3766 Wiggan Dr., Woodbridge, VA 22193 (US). **NORTON, James, J.**; Apartment 409, 1201 Braddock Place, Alexandria, VA 22314 (US). **CHENG, Meishan**; 2430 Dakota Lakes Drive, Herndon, VA 20171 (US). **PAPADOPOULOS, Ioannis**; 1517 Oronocco St., Alexandria, VA 22314 (US).

(74) Agent: **IM, Andrew, C.**; Fulbright & Jaworski LLP, 666 Fifth Avenue, New York, NY 10103 (US).

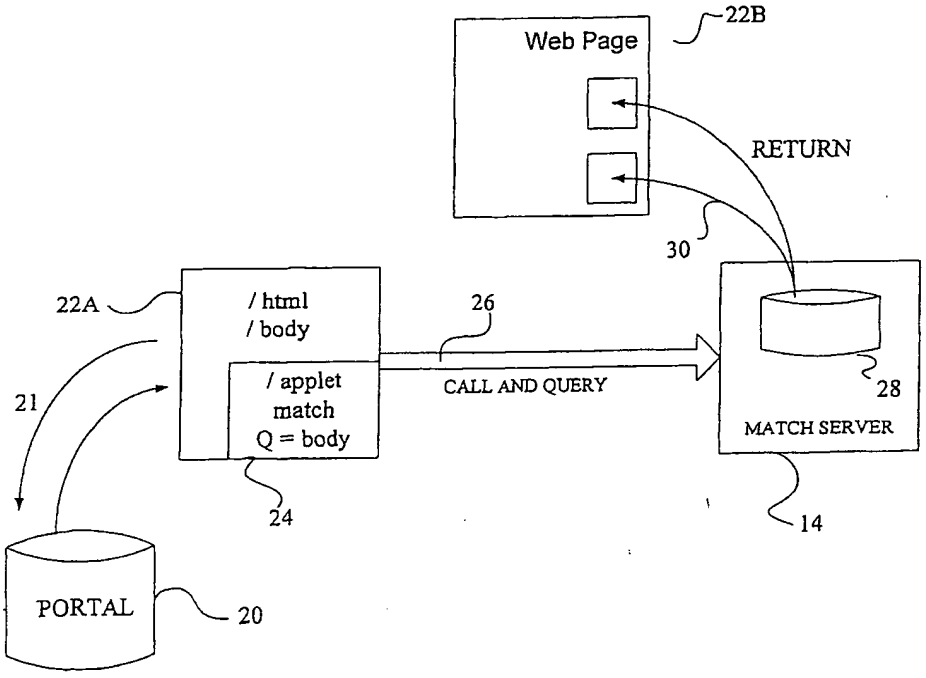
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*

[Continued on next page]

(54) Title: CONTEXT MATCHING SYSTEM AND METHOD



(57) Abstract: The system and method for automatically and contextually matching (14) offers to the content on a web page (22) that a user has selected in real-time. Additionally, the system and method further refines the resulting matches (14) based on business rules established by the web site (22A) owner or the system provider, thereby providing offers that are likely to generate selling opportunities to the web site (22A) owner.

WO 01/44992 A1

— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

CONTEXT MATCHING SYSTEM AND METHOD

Technical Field of the Invention:

The present invention relates to content analysis and, in particular, to correlating accessed data content with target content in Internet and wide network systems.

Background of the Invention:

Individuals and entities have certain interests and proclivities. Matching products or services with the interests and proclivities of individuals and entities is the goal of targeted marketing. The more accurate the match between the interests of the potential consumer and the offered product or service, the greater the efficiency and return from advertising expenditures. Consequently, targeted advertising can legitimately command higher prices than indiscriminate broadcast ad dissemination. For example, it is known that those who move into new houses are more likely to be purchasers of furniture than those who reside for years at a single address. Consequently, furniture stores often send advertising flyers to those with new homes. Similarly, those who live in more expensive neighborhoods are presumably more affluent and, therefore, more likely to be consumers of financial services than those residing in lesser neighborhoods.

Advertisements are often embedded in Internet web pages. Internet portals generate significant revenue by the placement of advertising on served web pages. To value such advertising, a variety of accounting methods are used. For example, ad cost can be correlated to a commitment to place a particular ad a certain number of times. Because such placements are often made randomly, valuations for such placements tend to decline as web "real estate" expands. Other, more sophisticated placement schemes rely on operator placement of categorical ads on the basis of search line contents. For example, if an Internet user searches for "Mount Everest" on an Internet portal search page, an operator can push ads related to mountaineering products into the returned pages. Such methods are slow and imprecise however, and

fail to contemplate any more than a gross sense of the user's interests and proclivities.

Search line content matching often fails to identify the goal of such searches. For example, the "Mount Everest" search may have actually been run by a professor of physiology with an interest in the effects of oxygen deprivation on mental acuity. For such a user, a more appropriate ad might be directed to medical equipment.

The sheer volume of material returned by a search request further blunts the accuracy of search-related correlation schemes. A particular search may return millions of items. Yet current systems do not resolve the mass of returned data into a focused perspective. What is needed, consequently, is a system and technique to associate Internet user proclivities and interests with suitable product and services offerings.

Summary of the Invention:

Therefore, it is an object of the present invention to provide a system and method which overcomes the short comings of the prior art system.

Another object of the present invention is to provide a system and method which automatically and contextually matches products, advertisements or other content (hereinafter referred to as "offers") to the content on a web page that a user has selected in real-time.

A further object of the present invention is to provide a system and method as aforesaid, which refines the matches based on business rules established by the web site owner or the system provider, thereby providing offers that are likely to generate selling opportunities to the web site owner.

Yet a further object of the present invention is to provide a system and method as aforesaid, which additionally refines the matches based on user profile information, thereby providing targeted offers that are likely to generate buying opportunities to the user.

A still further object of the present invention is to provide a system and method as aforesaid, which dynamically updates the user profile information in accordance with the user's responsive behavior.

The present invention matches document content with selected products or services and advertisements. A client accesses a matching system and gets a return of contextual matches for the content of a document under review or requested by the client. A client makes a call to a product matching engine through a JAVA applet, API, or other provided method and receives, in return, data related to products or services that match the contents of a client query, which is composed from the document. The matching engine receives a contextual query and an identifier and, in preferred embodiments, a parameter set which may be used to adapt the return to fit particular requirements and returns one or more products, images, services links, and links to buy the referenced items or access other data (hereinafter sometimes collectively referred to as "results" or a "results presentation"). The behavior is logged and incorporated by the matching system for use in subsequent exercises.

Various other objects, advantages, and features of this invention will become readily apparent from the ensuing detailed description and the appended claims.

Brief Description of the Drawings:

The following detailed description, given by way of example, and not intended to limit the present invention solely thereto, will best be understood in conjunction with the accompanying drawings in which:

Fig. 1 is a high-level block diagram depicting a preferred system embodying the present invention;

Fig. 2A is an example application of the present invention that uses a Java applet;

Fig. 2B depicts an alternative technique employed in accordance with the present invention to pre-match web pages with selected content;

Fig. 3 depicts an embodiment of the present invention;

Fig. 4 shows redirection based upon user behavior in response to matched and presented data;

Fig. 5 is a graphical depiction of a contextual query in the context of the preferred embodiment of Fig. 3;

Fig. 6 depicts the cache presence analysis of Fig. 3;

Fig. 7A depicts data retrieval in accordance with an embodiment of the present invention;

Fig. 7B depicts loading data in accordance with an embodiment of the present invention;

Fig. 8 depicts inventory or catalog loading in accordance with an embodiment of the present invention;

Fig. 9 depicts the process of a template in accordance with an embodiment of the present invention;

Fig. 10 depicts an update of a cache pool in a preferred embodiment;

Fig. 11 illustrates an intelligence engine or match server 14 in accordance with an embodiment of the present invention;

Fig. 12 illustrates an example of a contextual matching process of the present invention;

Fig. 13 is a system diagram of an embodiment of the contextual matching engine of the present invention;

Fig. 14 is a screen shot of the Business Rules Manager;

Fig. 15 is a screen of the Business Rules Manager for customizing business rules;

Fig. 16 is a screen of the Business Rules Manager for selecting promotional items; and

Fig. 17 shows a catalog integration method in accordance with an embodiment of the present invention.

Detailed Description of Embodiments:

The present invention is readily implemented by presently available communication apparatus and electronic components. The invention finds ready application in virtually all commercial communications and/or computer networks, including but not limited to world wide web (Internet), intranet, local area network (LAN), wide area network (WAN), wireless network and wired cable transmission

systems.

Turning now to Fig. 1, there is illustrated a high-level block diagram depicting a matching system 10 in accordance with an embodiment of the present invention. A client 12 communicates with a match server (or an intelligence engine) 14 through a signal pathway 16, such as the Internet HTTP protocol. It is appreciated that any signal protocol intelligible to both client 12 and server 14 can be used to operate as signal pathway 16 in the present invention.

The client 12 makes a call to the match server 14. The call from client 12 to match server 14 includes at least a contextual query, i.e., a document or a portion of the document. In accordance with an embodiment of the present invention, the call includes a parameter set (or business rules) to pare the returned data set by the match server 14. For example, the client 12 can establish the parameter set to receive results that are relevant to its business requirements or needs. Preferably, the call can include an ID for verifying that the client 12 is authorized to access or communicate with the match server 14. If the match server 14 determines that the ID associated with the client 12 is valid (i.e., the client 12 is authorized to access the match server 14), the match server 14 returns to the client 12 a set of matched data or offers, including but not limited to product and service descriptions, advertisements and/or associated links. In accordance with an embodiment of the present invention, the associated links can include buy links to enable the client's customer to view and/or purchase products or services that are contextually related to the query.

Turning now to Fig. 11, there is illustrated an intelligence engine or match server 14 in accordance with an embodiment of the present invention for contextually matching data to content using business rules. The match server 14 goes beyond picking up key words from the document, the match server 14 "reads" or analyzes the full context of articles or information using a sophisticated layering of artificial intelligence, neural networking, linguistics and natural language processing.

The match server 14 conducts the same analysis of the inventory of offers (such as products, services, advertisements, etc.) available to be matched to that content. The offers are then ranked in order of relevance to the meaning of the article

and the most relevant offers are dynamically served right within the content on the Web page. To the site visitor or user (i.e., customer of client 12), the process is seamless.

For example, if a visitor or customer of client 12 selects an article about BMW's newest vehicle, the match server 14 of the present invention presents a buying opportunity to the customer, i.e., an attractively discounted book for "BMW Enthusiasts". If the customer is reading an article about Tiger Woods, the match server 14 displays or presents items such as Nike golf balls or a book on the history of the US Open to the customer. It is appreciated that the intelligence engine 14 of the present invention understands the difference between Tiger Woods, a tiger, and woods.

In accordance with an embodiment of the present invention, the intelligence engine 14 comprises at least a contextual matching engine 210 and an entity extractor 220 for extracting entities, such as names of a person, company, organization, place, product, or the like. Through the application of business rules 240, the intelligence engine 14 finds offers that are relevant to the needs of the client 12, i.e., a web site owner. The business rules 240 are behavioral constraints applied to matches 230 found by the entity extractor 220 and the contextual matching engine 230 as a result of a query 200. That is, the intelligence engine 14 combines the results from the contextual matching engine 210 and the entity extractor 220 into a single raw, unfiltered list of products 230 not conforming to any business rules. By applying the business rules 240 to the raw list 230, the intelligence engine 14 generates a new, more targeted list 250, which is presented to the client 12. A user profile could even be incorporated into the business rules 240 to further tailor the targeted list 250 for each specific user. The targeted product list 250 is a subset of the list 230 that has been filtered through the business rules 240.

Through the application of business rules 240, the intelligence engine 14 provides the client 12 with more focused and relevant matches. Preferably, the intelligence engine 14 applies the constraints of the business rules hierarchically. For example, the visitor's selection of an article or document may result in a search on the

general topic of "business" being performed by the intelligence engine 14. The intelligence engine 14 may apply the business rules 230 associated with the client 12 to narrow the search further to "books related to business," thereby focusing and refining the search to target the needs of the client 12. In other words, the client 12 may be a book seller utilizing the match server 14 to generate selling opportunities. In accordance with an embodiment of the present invention, the intelligence engine 14 may recursively apply the business rules to the results of the contextual matching engine 210 and the entity extractor 220. In the book example, the intelligence engine 12 can further refine the results by applying a second business rule limiting the "books" category of the "business" to a sub-category of "best sellers." The products 250 returned by the intelligence engine 14 then reflect the constraints of "business," "books" and "best sellers." Whether the business rules 240 are set by human definition or via an automated application of a Baysian network intelligence, the returned matches 250 are the product of relevancy scoring leading to the products being targeted to the individuals in a more refined manner.

By narrowing or focusing the query 200 through the business rules 240, the intelligence engine 14 can return matches that are more relevant (i.e., higher return values) to the client 12. That is, more specific filtering by the intelligence engine 14 results in matches that have higher return values, thereby enabling the client 12 to present relevant products to their customers. Accordingly, the business rules 240 provide the client 12 with more selling opportunities because more relevant offers (i.e., products, services, advertisement, etc.) are presented to its customers.

Turning now to Fig. 12, there is illustrated an example of contextual matching process in accordance with an embodiment of the present invention. The inputs 400 comprise both the content that a web site visitor has selected, i.e., the query 200 (Fig. 11) and the offers to be matched to that content. The web site visitor may represent a customer of the client 12 while offers may represent products, services, advertisements, promotions, data or other content. The client 12 delivers the content from the web page to the intelligence engine 14 automatically through web page mark-up language that the web site owner placed on each page. The web site owner

can place the contextual tags on every page in its web site or can selectively target pages to display contextually targeted offers. It is appreciated that the contextual tags represent queries 200 or can be converted into queries 200 by the intelligence engine 14. The offers to be contextually matched to that content or queries 200 are sent from the database to the intelligence engine 14, where they are then categorized and stored. A detailed description of the process of submitting product data is described herein.

In accordance with an embodiment of the present invention, an API can be used to place the contextual tags on the web to display contextually relevant products and special promotions with minimal human intervention and effort. A plurality of matched products or items can be displayed, e.g., two contextually matched products, two products from the entity extractor 220 (Fig. 11), and two promotional/default products or items. Preferably, the results presentation 260 (Fig. 11) can be selected from a library of templates or can be customized by the client 12. The results presentation 260 can support multiple formats, such as HTML, Java, etc. It is appreciated that each format may support slightly different functionality.

For example, the functionality of the Java-based results presentation 260 can include the ability to automatically (every n seconds) or manually rotate from one product to the next and back, and also the ability to click on the product offering directly or on the "Buy/More Info" button to go to the purchasing pages of the client 12, i.e., the web site owner, or a merchant. Comparatively, the functionality of the HTML-based results presentation 260 is the same as the Java-based results presentation 260, but the HTML-based results presentation does not provide the rotation capability. Instead the HTML-based product presentation 260 provides the ability to display a random choice of the matched products, services, advertisements, promotions, data, etc., offering diversity to the viewer, i.e., the web site visitor.

Fig. 2A illustrates an example of an applet application in accordance with an embodiment of the present invention. An applet 24 makes the call 26 to the match server 14. A portal 20 can represent any Internet portal, search engine, directory server or database such as Yahoo®. The portal 20 responds to a page request 21 from

a browser 22 as directed by the web site visitor. It is appreciated that in this example, the browser 22 functions as the client 12 (Fig. 1). The browser 22 is shown sequentially as browser 22A running the requested page as a series of HTML tags. The returned page has embedded applet 24 therein. The browser 22 loads and runs the applet 24 that includes a query 200 (Fig. 11) defined to be the body of the page, i.e., contextually representative of the page. The applet 24 issues a call and query 26 to the match server 14, which performs a match on the body of the page. The match server 14 performs a content analysis of the query, i.e., on the body of the page, and selects from content database 28 a set of matched return data 30. The match server 14 determines the matched return data 30 in accordance with its internal rules, i.e., contextually matching, entity extraction, etc., and any parameters or business rules 240 included with the call 26. A relevancy score can also be returned for operation on by the client or, alternatively, the relevancy score can be used internally by the match server 14 to pare the return data 30 to comply with any business rules 240 conveyed with the call 26. The results are placed in the web page as opened by the browser 22 as shown in state 22B. In accordance with an alternative embodiment of the present invention, the call and query 26 can be directed to a dynamic image generator (not shown) of the match server 14 which dynamically returns the matched information as a graphical image, e.g., a .gif file.

For a Java applet, the following sample html codes can be embedded into the body of the client's web page where the product presentation 260 is to appear:

<applet code=v3gui codebase='http://cm.yellowbrix.com/v3/' width=x height=y>
<param name='ClientId' value='Your partner ID'>
<param name='query' value='Text to be matched goes here'>
<param name='pageId' value='A code used for tracking. Max 10 characters'>
</applet>; or

<APPLET code='simplegui' codebase='http://cmatch.yellowbrix.com/match/code/gui'

Code	What it is Doing
<APPLET code='simplegui'	The code corresponds to the name of the applet.
Codebase='http://cmatch.yellowbrix.com/match/code/gui'	The codebase corresponds to the server location where the program resides.
width=160 height=250>	The width and height are parameters used by the browser to size the applet window on the page.
<PARAM name=query value='[ENTER QUERY DATA TO BE MATCHED HERE]'>	The PARAM indicates the parameter of the information used by the applet. In this case, it is matched to products that are displayed in the applet window.
</APPLET>	The </APPLET> tag indicates that all of the information necessary to run this applet has been given. This closes the code for the applet.

In the alternative technique shown in Fig. 2B, the client 12 is the site owner. Through API 32, the client 12 calls the match server 14 through a call 34 which can include a parameter or business rules 240 as described herein with reference to Fig. 2A. The match server 14 performs a contextual analysis and matching exercise with the documents provided with the call 34. The match server 14 performs the contextual analysis using the content database 28 and returns pages in which matched data is embedded as depicted by a return 36. The embedded pages are returned to the client 12 for storage in a database 38 and for publication to the browser 22 as static pages upon request from a user, as depicted by a publication arrow 40. The browser 22 displaying the page with embedded data represented by reference numbers 42A and 42B is shown as browser 22C. A variable set of customizable parameters may be passed through the API. For example, parameters either in API or other modes such as applets, for example, may have parameters such as:

Number of results to return (min, max)

Type of result (HTML, Java)

Query (Context to be matched)

Client side integration typically involves embedding the call to the API within the pages served by the client web server or including a call to the API from the client page generation engine to embed results before the page is served.

Some exemplar parameter definitions include:

Id = The client will be setup with a registered id to use when requesting calls through the API for products. Unless this id is already confirmed, the API will immediately return with 0 results.

Embedded use:

<param name="id" value="clientid">

Integrated use:

&id=clientid

Profile = business rules selected by the site owner

showNum = The number of results can be controlled by the client with this parameter. The default value is 1 and the maximum is 3 (currently).

Embedded use:

<param name="showNum" value="3">

Integrated use:

&showNum=3

query / rmquery = The client must define this variable with the content they wished to be matched to products. Specifying no query will return in a random list of products. Limitations are dictated by whatever their servers are set to send through the HTTP protocol and other browser limitations. For best results, the query should be kept under 255 characters and be set with pertinent information.

Embedded use:

<param name="query" value="Latest Release of Mariah Carey's Album takes off">

Integrated use:

&rmquery=Latest Release of Mariah Carey's Album takes off

version = The version values (currently) are 0 and 1. 0 sends the results back in HTML format and 1 returns the results back also in HTML format but defines the use of an applet once the page is loaded through the web server.

Embedded use:

none

Integrated use:

&version=1

For non-Java output, the following sample API call can be used:

BASE PARAMETERS(http://cm.yellowbrix.com/v3/v3api.nsp):

id= valid partner id

pageId= A code used for tracking. Max 10 characters

version=

{

raw, #unformated output

html, #templated html output based on 'id'

"random=" parameter; If random=1,

output will change on reloads,

otherwise the first product is

returned.

"bgcolor1=" HEX parameter; Color for the top/bottom bars

dhtml, #dynamic html

"delay=" parameter; in microseconds, the time each product

remains on screen. Default is 3000 i.e. 3 secs.

flash, #output in flash generator format

xml, #extended markup language

java, #java data, not java applet(see v3gui)

}

query= URL encoded text to be matched.

Returning to Fig. 12, the analysis and matching process all takes place in the intelligence engine or match server 14, where the inputs are processed for entity identification and contextual analysis. Preferably, the meaning, or "context" of each piece of data is used to generate neural net vectors, which are sometimes referred to as feature vectors (feature vectors is used an encompassing term that includes all types of vectors). The vectors from the web site content are then compared to the vectors associated with the inventory, and the resulting matches are scored and ranked. The contextual matching engine 210 (fig. 11) provides the necessary facility for automatically matching arbitrary textual content to inventory, thereby making it possible for the clients 12 or web site owners to offer results more closely relevant to their site content. With the contextual matching engine 210, a web site owner is able to control the matching results. As shown in Fig.13, in accordance with an embodiment of the present invention, the contextual matching engine 210 is composed of three subsystems: (1) the contextual matching server300, which handles client requests and contains the logic for the system; (2) the vector generator 310, which generates document vectors; and (3) a relational database 320, which stores the inventory information. The contextual matching server 300 accepts connections on an established port. These connections include, but are not limited to, a text entity (a paragraph, abstract, entire article, etc., specifically on a single topic) to be matched, a metadata query to satisfy, and/or the number of desired returned products.

It is appreciated that the results were grouped into clusters of related results in preprocessing and then stored in the relational database 320. The contextual matching

server 300 generates a query context vector, or feature vector, using a vector generation algorithm. Generally, vector based generation algorithms have certain features in common: (1) they all characterize documents based on the presence of keywords; (2) they all associate vectors with these keywords; and (3) they all form document vectors by combining the vectors of the keywords present in the document.

According to one embodiment, the present invention provides a novel approach to representing textual documents as high dimensional vectors. Such an approach provides an efficient means of indexing document collections, allowing retrieval of documents (querying) based on keywords, grouping related documents (categorization). Additionally, this method supports such conceptual queries and document groupings. This method of obtaining document vectors via Explicit Term Co-occurrence (hereinafter "ETC") proceeds as follows: (1) A set of documents is selected (the corpus) for use in training. (2) The corpus is stemmed, meaning that every word in the corpus is converted into its root form. For example, the word "programmer" is stemmed by the word "program." There are several algorithms in the literature for performing stemming, e.g., Porter's algorithm. The method employed in ETC is an aggressive one known as "derivational stemming" which utilizes known word morphologies and greatly reduces the overall stem set for the corpus. (3) "Stop words," which are words that are known to convey little or no semantic value (e.g., articles such as "a", "the", etc.), are removed from the corpus. (4) Every unique stem word in the corpus is identified, and the frequencies of occurrence of these words are computed. (5) A list of keywords is compiled from the stem words and then arranged from the most frequently appearing stems to the least frequently appearing stems. This list can be chosen to be of any desired size, or a metric system based on the upper and lower frequencies can be used. (6) A term/document matrix is computed. The rows of this matrix represent the keywords in our list and the columns represent the documents in the corpus. The values at each position are "weights" for each keyword with respect to each document. The weights used in ETC are simply the number of times each keyword appears in each document. For example, given a simple corpus consisting of two documents and a keyword set

consisting of two stem words, one might obtain the following term/document matrix:

| 1 2 |
| 0 1 |

This matrix indicates that the first keyword appears once in the first document and twice in the second, while the second keyword does not appear at all in the first document and only occurs once in the second. In practice this matrix will be much larger, as the corpus generally consists of many documents (this number is denoted as N) and the keyword set will consist of many keywords (this number is denoted as M). The resulting term/document matrix will therefore have MxN elements. (7) Using the term/document matrix, a term co-occurrence matrix is computed. This matrix represents the number of times in the corpus, pairs of words occur in the same document. Unlike the term/document matrix, the columns and the rows represent keywords. Hence, the matrix is MxM. For example, from the term/document matrix above, the following term co-occurrence matrix is generated:

| 3 1 |
| 1 1 |

This matrix is interpreted as follows: the first term occurs with itself three times, the first and second term occur together once, and the second term occurs with itself once. Notable properties of the term co-occurrence matrix are as follows: it is symmetric about the main diagonal and values on the main diagonal represent the total number of times a keyword appears in the corpus. Another useful property not evident in this example is that this matrix tends to be sparse, that is, most of the values are zero, for typical keyword sets. This
is a consequence of the fact that most terms do not appear in conjunction with most of the other terms. Sparsity allows for compact storage using one of many sparse matrix representations known from the literature. (8) Unit length vectors are then generated for each keyword using the term co-occurrence matrix. This is accomplished by computing the eigen values of the co-occurrence matrix and their associated eigen vectors. The eigen vectors associated with the K largest eigen values ($K << M$) are

used as column vectors to form a new MxK matrix. The rows of this matrix are then normalized to unit length, and are used as the K dimensional vectors for the keywords. Appropriate values of K can be determined empirically or by computing the ratio of the Kth largest eigen value to the largest eigen value and choosing the value of K that provides the desired ratio. (9) For a given document, a vector is generated by taking a weighted sum of the keyword vectors for the keywords appearing in the document:

$$Vd = \frac{(w1 * V1 + w2 * V2 + \ldots + wn * Vn)}{\text{sqrt}(w1*w1 + w2*w2 + \ldots wn*wn)}$$

where w1 ... wn are the weights of each term with respect to the document and V1 ... Vn are the keyword vectors. The weights can be computed using any of several rules, and represent the relative frequency of the terms in the document.

After the vector generations, the document vector, or feature vector, must be compared with database vectors, or feature vectors. A naïve approach to product matching would be to compare the document vector to the vectors for every product in the database. This process becomes more burdensome as the number of products in the database grows. Fortunately, the database vectors are not smoothly distributed throughout the vector space, but rather, tend to "clump" together, leaving vast empty spaces between the clumps, or clusters as they are commonly known.

For any given cluster, there is a sphere which bounds every point in the cluster. For all the various clusters, one could compute the center point and the radius of the sphere which bounds the cluster. Then, when one wishes to find the products that match a given document vector, one need only compare the document to the products in the cluster whose bounding sphere contains the document vector (or the nearest spheres, if no sphere contains the document vector). Thus, the computation is reduced from comparing the document vector to all the product vectors to simply comparing the document vector to the center vectors for the spheres, followed by comparison to the product vectors for the products in the matching spheres.

In practice it is difficult to compute the bounding spheres directly. This invention provides an improvement over the prior art by computing an approximate set of spheres. For the CME, the following steps are employed: (1) The process begins with no spheres. (2) Then, for each product, Pi, if the product vector, Vpi, is contained in an existing sphere, that product becomes associated with the sphere. Otherwise, a new sphere is created, Sj, centered at Vpi with radius R, where R is chosen empirically. Pi is then associated with Sj. This procedure is sub-optimal in that it generally produces more spheres than are necessary to bound the clusters (partially do to the fixed radius, R). More importantly, the distance between spheres may be small. Because of this, the above matching algorithm is modified slightly. Rather than just taking the sphere that contains the document vector, one must always choose the N spheres closest to the document vector. N is determined empirically to provide the best compromise between match quality and computational burden.

The contextual matching server 300 then compares the query context vector to the pre-determined item context vectors to narrow the search to focus on products that are most likely to be relevant to the selected text. In accordance with an embodiment of the present invention, the query context vector, is compared to the center vector of each cluster and the clusters with center vectors closest to the query context vector are selected. It is appreciated that these clusters IDs are then used to narrow the scope of products returned by the original metadata query as issued by the client 12. In other words, this narrowing qualification is added to the original metadata query to form an SQL query. Preferably, the contextual matching server 300 returns N most relevant to the client or the E-commerce applet 12, along with their associated relevance.

Preferably, the user can, as an option, set the matching process to return results based on the default matching, or based upon user-defined business rules 240. If the user selects results based on general topic matching, then the system will employ the predefined (default) business rules 240. Whereas, if the user selects results based on business rules 240, then the system will pass the text to the vector generation subsystem 310. The vector generation subsystem 310 employs user-defined business rules 240 to return results that are filtered, i.e., biased towards the business rules or

instructions 240.

In accordance with an embodiment of the present invention, the business rules 240 are applied on the front end of the contextual analysis process to define the inventory sources to match to and the minimum relevancy score acceptable for contextual matches. The business rules 240 are applied again after the contextual analysis to sort the relevant matches based on predefined priorities, such as profit margin, release date and best sellers. These rules then define the display criteria, set promotions that should be served in addition to contextual matches, and set default offers when no contextual matches were found to meet the site criteria. Preferably, the profile-based personalization, i.e., process of learning the user's buying habits, can also be integrated into the contextual matching process of the present invention.

Once the contextual matches have been made, the related product information is embedded in the contextual tag and dynamically delivered back to the site. These matches are now cached on that page so each time a site visitor calls that page the relevant offers are dynamically served within the page. Whenever the page content, inventory of offerings or business rules are updated, the content is resent to the intelligence engine 14 for new matches.

Through this online GUI the client site can view detailed reports to see their results and optimize those results by applying their market and merchandising expertise to edit or build new business rules 240, set promotions, and if desired, supervise the matches.

In accordance with an embodiment of the present invention, a dashboard 410 is provided to help the users, such as the web site owners, manage their accounts and retrieve reporting information. The dashboard 410 is comprised of three main components: a business rules manager, a report manager and an inventory manager. Preferably, a tool for Supervised Matching is also provided.

The Report Manager allows users to view all of the key metrics for their account, such as measures for calculating contextual matches, promotional matches, and an aggregate of both contextual and promotional Matches.

Reporting Measure	Definition

Contextual Matches	
Avg. Relevancy Threshold	For the selected time period, the average relevancy threshold that was set by the user.
Matched Pages	The total number of pages processed that resulted in a contextual match – not impressions served.
Match Ratio	The percentage of paged processed that resulted in a contextual match (total processed/total matched)
Avg. Match Relevancy	The average relevancy score for the products matched on pages with contextual matched.
Impressions	The number of contextually matched products served as an impression.
Click-throughs	The number of products served that was clicked through for more information/buy.
Click-through Rate (CTR)	The percentage of matched products served compared to those that were clicked through for more information.
Avg. CTR Relevancy	This is a weighted score indicating the average relevance of the click-through products in comparison with the contextual meaning of the article.
Sales	The number of products sold
Sales Conversion Rate	The percentage of contextually matched products sold from click-throughs (sales/click-throughs)
Avg. Conversion Relevancy	The weighted average relevancy score for all products sold.
Promotions	
Promotional Products Served	This number indicates the total promotional products (impressions) that were served.
Click-throughs	The number of promotions that were clicked through for more information.
Click-through Rate (CTR)	The percentage of the promotions served that was clicked.
Sales	The number of promotional items served that was purchased.
Conversions	The percentage of promotional items served that was purchased.
Aggregate	
Impressions	This number is the total of pages with matched and/or promotion served. (Note: This is not the sum of matched pages and promotional pages served as both matched and promotional products can be served on the same page.)
Total Click-throughs	This number is the total of matched products and promotional products clicked.
Average CTR	This number represents the total amount of click-throughs divided by the total number of products served.

Total Sales	This number represents the total number of products sold from both contextual matching and promotional sales.
Average Sales Conversions	This number represents the total number of sales divided by the total number of click-throughs.

The default for reporting will be the past 30 days. There will also be an option to the users to select the date range they want to view and recalculate the figures. The reporting can be modified to provide an overview of the performance of the entire site based upon a defined Business Rule or based upon a specific page that is defined by the users.

The profile manager allows the users to access information about their profile configurations. For example, profile can be made up the following elements:

Element	**Definition**
Rule Name	This element is the name assigned to a set of profile configurations. Every account has a Default profile that cannot be deleted.
Inventory Provider	This drop down menu lets the users select the source of the inventory to be used for the profile. The default is ECS.
Category	This drop down menu lets the users specifically define what types for products from the inventory provider they wish to display. The default value is "no constraints."
Minimum Relevancy	This number is used to correlate the relevancy of a product to the context of a page. This is number can be scaled from 1 to 100 with 1 being the most relevant and 100 the least relevant.

If the users select the Business Rules Manager from the Dashboard 410, they will be able to configure the commerce offerings for their site. The users can select one of the preconfigured Business Rules selections, or design a tailor-made business rule for themselves.

For example, a screen shot of the Business Rules Manger is illustrated in Fig. 14. When the user clicks on an existing Business Rule, the merchant, products and promos for that rule will be displayed in the text box to the right. If the users want to select the highlighted rule, they can click the **Apply** button and it will take them to the page with the look and feel configurations. For users who do not select one of the

predefined Business Rule sets, they can configure customized Business Rule by click the **New Rules** button.

For example, a screen shot the Business Rules Manager for customizing business rules 240 in shown in Fig. 15. In order for customers to define a Business Rules set 240, they must first enter a name for their configurations. After they have entered a name in the designated text box, the users can select different Product Categories from the available merchants. Each merchant will have the applicable Product Categories in a separate column.

These categories are designed to be options so there are duplicate categories among merchants. Each category has a corresponding subcategory select box. The selection box allows the users to choose more than one subcategory for each corresponding category.

After all of the desired subcategories are selected, the users can enter the Relevance score. This is a percentage that determines the minimum relevancy score of the product to the context of the content.

The final configuration for the Business Rules set is the determination of the sort order. The users can rank the priority of Relevance, Price and Best Seller for the products they have selected. Since there are three sorting selections, the rank order is 1, 2 or 3 with 1 being the highest rank. If the user enters the same number in more than one drop down, an error message is received.

This ranking can be done for each individual category, but applies to all products within the selected subcategories. If the users do not select a sort order, they can leave the drop down menus at the blank default.

After the rankings have been selected and all other information has been designated, the users click the **Next** button. This will open a summary screen for the users to review the selections they have made for all vendors in a summarized form that can be printed for their records.

For example, a screen shot of the Business Rules Manager for selecting promotions is shown Fig. 16. When all the inputs have been approved, the users will be taken to the Promotions section of the Business Rules Manager. The promotions

will be limited to the categories that were selected by the users in the pre-generated and customer profile configurations.

The users select a product category from the drop down list and enter a keyword to narrow the search. The users click the **Go** button and a popup window will be show the available promotions for the category and search criteria.

The users will check the boxes for the promotional items they want to add to the business rules profile. When they have selected all the desired items, they click the "**OK**" button and will be returned to the Promotions page.

If the users do not want any of the items, they can click the "**Close this window**" link and returned to the Promotions page. When the users have completed all the promotional product configurations, they will click the "**Next**" button to select the look and feel of the presentation box 260.

Turning now to Fig. 3, there is illustrated another embodiment of the match server 14 of the present invention. The client 12 generates a call and query 44. The authentication at ensuing process block 46 is the first process at what has been previously shown as the matching server 14. If authentication of the client or user ID fails, an exception is generated and the process ends as depicted by process block 47. If authentication is successful, the contextual query, e.g., the body of a web page, is forwarded to an analysis process shown as process block 48. At process block 48, the query is analyzed as being either text or binary. A text query is simply a word or series of words such as sentences or paragraphs. A binary query is composed of images, sound, compiler code, executable code, compressed data or other similarly binary representational data. The query is coded for comparison purposes. It is appreciated that a variety of methods can be used to code the query data, such as, checksum.

A cache 50 is examined to determine if data corresponding to the checksum or code has previously been matched by the match server 14 and, if so, the page is immediately sent to the client 12 on return line 51. Preferably, the request and the matched data are logged in a database log 56. If the checksum or code is not found in the cache, a call is made to relevancy process 54 for analysis of the query. A variety

of known techniques for relevancy scoring can be used, such as Resource Miner.

The relevancy process 54 determines a relevancy score based on any parameters or business rules 240 from client 12 or other rules generated internally in the match server 14. The rule set is depicted as process block 52. Parameters could include rules or requirements such as the number of books or cars to be matched or, the parameter list could include either with or without literal numeric limits, a relevancy score limit from which only those found matches that achieve a relevancy score greater than x are to be returned.

Match selections from relevancy process 54 may also be pared by the rule set at process block 52 to sculpture the return to known or learned data concerning the user by the client 12. For example, previous buying habits of the user or customer of the client 12 stored in the database log 56 may be contemplated in the output from relevancy process 54. Alternatively or in addition, categorical inclinations such as demographic propensities may be contemplated to return items of likely interest to those in the demographic group of the user for the client 12. Changes in user interests as exhibited in previously logged web page access or purchase decisions may also be used to modify the rules to adapt to changing interests of the user at client 12. Observed factors such as page dwell time may further refine return data to correspond to user interests and inclinations.

Internal rules may be devised in conjunction with the database log 56. Any time a query is made, logging activity is executed at the database log 56 to allow the match server 14 to "learn" about the efficacy of the scoring system in use by the relevancy engine as well as "learn" about the user and those similarly situated. To that goal, when a request is made or a query-matched product reference is returned to the client 12, or a product "buy link" is clicked, the transaction is logged in the database log 56. Other information, such as a click from client 12 on one of a returned set of product references is also logged in the database log 56. This aggregated data allow internal analysis of the system and the creation of a rule set internal to the match server 14 to increase efficacy. In accordance with an embodiment of the present invention, the relevancy process block 54 can include a

relevancy engine along with analytical tools devised to contemplate data from the database log 56.

After the query is filtered at the process block 52, the contextual database inventory block 58 is searched for relevant matches. The database 58 is selectively loaded by catalog load 59 according to business relationships between the owner of match server 14 and the web site owner who wish to place or display the product or services listing. The database 58 can also be selectively loaded to fit any dimension such as climate (i.e., winter, summer), season (i.e., Christmas), demographics (retired, generation X), financial status or other overlay to business contexts.

After matches are selected and filtered using business rules and/or user demographic information, the template formatter 60 fits the return material in the returned page. The cache 50 is updated through an update cache 62 and the transaction is logged in the database log 56 and the return page is sent to the client 12. As previously mentioned, responsive behavior such as logging on a "buy" site is logged from the click capture line from the client 12, shown as line 64 in Fig. 3.

In accordance with an embodiment of the present invention, the following steps are involved in implementing the matching system of the present invention:

1. Content Criteria: The client, i.e., the web site owner, determines the content criteria, such as inventory, and product matching criteria. The product inventory can come from merchants, provider of the match server 14 and/or affiliates.

2. Contextual analysis: Product descriptions are filtered, analyzed, and indexed in advance. Neural network techniques are used to identify concepts in product descriptions, and duplicates with different media and product families are identified (i.e.: paperback vs. hardback, items of different color, etc.).

3. Matching: The body of the text of each web page is used as the query. Neural network and linguistic pattern techniques are used to identify contextual matches in real time, and the context of the content is matched to the context of the inventory.

4. Matching refinements: Boolean filters, user profiles and established

business rules are used to refine contextual matches.

5. Storage: Standard database technologies are used to organized inventory, to index products in advance for performance, and to rapidly implement the matching processing systems for clients.

6. Result serving: Matches can be placed on any page, on any site. As pages are served, appropriate products are delivered automatically. Results are presented in a number of convenient ways, such as alongside content as HTML, a gif, java or java script or as a pop-up.

7. Integration: The matching processing system of the present invention works with any content, any product inventory, any user profile, any Web site, and it can link/interface with any transaction engine

Figs. 3 and 4 combine to show redirection based upon user behavior in response to matched and presented data returned to client 12 on return line 51 (Fig. 3). As depicted by process block 64 (Fig. 4), once client 12 is returned with a requested list of products and services, options for user-interaction such as loading, reloading, URL links 66, viewing and stopping, etc. are redirected through match server 14 for behavioral analysis and tracking of user actions. The results of such analysis are deposited in database log 56. These statistics are used to formulate behavioral trends on how, when, and where the list of returned results are used, and to provide financial analysis in the case of transaction or fulfillment reporting.

Fig. 5 is a graphical depiction of the contextual query of process block 44 of Fig. 3. The query ID and list of parameters are analyzed. ID 70 and the context of the query itself condition the entire contextual analysis. If no other parameters are with the query, they may be set to default values P_1, P_2 and P_N can be null or, if desired, restrictions used, as client set parameters are used, as elements of a BOOLEAN AND with the context.

Fig. 6 depicts the cache presence analysis of process block 50 of Fig. 3. The query's checksum 74 is used by process block 76 to lookup through DB lookup 76 an existing list of results already present in cache pool 80 that match the coding results from contextual query analysis 48 (Fig. 3).

If an appropriate list exists, those results associated with the list are found as indicated by process block 78 and returned to client 12 as indicated by block 82. The request and return are logged in the database log 56. If not appropriate list is found in cache pool 80, the query is sent on to the relevancy matching process as indicated by process block 84.

Fig. 7A depicts retrieval from contextual database 58. After filtering through filter process 52 shown in Fig. 3, the now filtered query is depicted at block 86. A call 88 is made to the contextual inventory and raw data results 90 are returned to relevancy process 54.

Fig. 7B depicts loading data for passage to template formatter 60 of Fig. 3. Raw data 90 of Fig. 7A is normalized at relevancy process 54 as indicated by block 92. It is then stored in temporary storage 94 by loader process 96 until a return set is completed. The entire temporary storage 94 then replaces the current database for production and update as shown by block 98.

Fig. 8 depicts inventory or catalog loading of process 59 of Fig. 3. As shown in block 100, inventory providers send raw formats of items for inclusion. Preferably, the inventory providers can select the frequency of updates to keep the data current. The components are identified and marked or coded at block 102 and sent to process block 104 for normalization and process blocks 106 and 108 for data integrity verification. It is appreciated that the components can represent merchant name/id, category name, URL, and the like.

Fig. 17 illustrates a catalog integration method in accordance with an embodiment of the present invention. The process starts with the catalog or inventory. To quickly serve a matched product, each product is classified by the specific category to which it belongs. Once a product has been categorized, it is then grouped into a catalog. A category is a specific, individual description of a product. A catalog contains different categories of products.

For example, Palm Pilots in the Hand-held Electronic Organizer category. The Hand-held Electronic Organizer category is part of the Computer catalog. The inventory is stored within a database using this architecture. For the database to define

other attributes about each product, the catalogs have to provide the following information:

ID, NAME, DESCRIPTION, CATALOG, CATEGORY, PRICE, COST, QUANTITY, URL CONSTRUTION (i.e. link to more information and buy link)

Most of these fields are used at display time, but there are the mandatory fields such as the ID, NAME, DESCRIPTION, CATALOG and CATEGORY that help keep track of each product. Once the information is stored in a database, it is ready to go through the context-training phase. This phase is where each product is analyzed and stored alongside its meaning.

The database is queried one product at a time and fed through a neural network to produce a context vector. After its meaning is produced by a neural network, it is then stored within the neural network's own database. At this point, we have a complete cognition of each piece of the entire inventory.

After the context-training phase, the internal structure is in place. The next phase involves developing an API that will wrap around the neural network database of products for retrieval. The API handles the request from a client, re-routes it to the neural network database and analyzes it to match the appropriate products that relate to the initial query.

In addition, the API is responsible for checking for data integrity of both incoming and outgoing traffic, caching certain data for optimization and providing statistical data for reporting and re-tuning procedures.

Fig. 9 depicts the process of template formatter 60 of Fig. 3. As inputs template, formatter 60 takes in raw data 110 and formats specific parameters 112 which can either be pre-loaded or conveyed to match server 14 with a query. The format parameters are components of style dimensions 114, 114_1, 114_2, 114_3, 114_N. The resultant return set and request is logged in the database log 56 and returned to client 12 on return line 51.

Fig. 10 depicts the update of cache pool 80 of previous Fig. 6. When the

cache is updated, the code or checksum is calculated and a key value pointing to the associated list of products 118. The key value and list of products are added to cache pool 80.

In accordance with an embodiment of the present invention, programs written to route the client 12 requests, analyze and filter data are written in Java and C++. Java is used in the match server 14 with the Java Servlet standard. This provides quick start threading and improved scalability to enable a round robin server when loads increase. This reduces development time and ensures a reliable mechanism.

Although the present invention and its advantages have been described in detail, it should be understood that various changes, substitutions and alterations can be made herein without departing from the spirit and scope of the invention as defined by the appended claims. Moreover, the scope of the present application is not intended to be limited to the particular embodiments of the process, machine, manufacture, composition of matter, means, methods and steps described in the specification. As one of ordinary skill in the art will readily appreciate from the disclosure of the present invention, processes, machines, manufacture, compositions of matter, means, methods, or steps, presently existing or later to be developed that perform substantially the same function or achieve substantially the same result as the corresponding embodiments described herein may be utilized according to the present invention. Accordingly, the appended claims are intended to include within their scope such processes, machines, manufacture, compositions of matter, means, methods, or steps.

WHAT IS CLAIMED:

1. A method for contextually matching offers, comprising the steps of :

selecting a content by a user;

contextually analyzing said content to determine the context of said content; and

selecting offers from a database matching the context of said content.

2. The method of claim 1, further comprising the step of filtering said offers in accordance with a business rule to provide filtered offers.

3. The method of claim 2, further comprising the step of ranking said filtered offers based on relevance to said content.

4. The method of claim 1, wherein said offers in said database being arranged into clusters, each cluster having pre-determined feature vectors and comprising contextually related offers; wherein the step of contextually analyzing comprises the steps of generating a feature vector from said content and comparing said feature vector to said pre-determined feature vectors to determine contextually related clusters; and wherein the step of selecting selects offers from contextually related clusters.

5. The method of claim 4, wherein said each cluster further comprising a center vector; wherein the step of comparing compares said feature vector to center vectors to determine at least one contextually related cluster; and wherein the step of selecting selects offers from said contextually related cluster.

6. The method of claim 5, wherein said center vector of said contextually related cluster being closest to said feature vector.

7. The method of claim 1, wherein each offer being at least one of the following: product, service, content item, advertisement and promotion.

8. The method of claim 1, wherein the step of contextually analyzing includes the step of extracting entities from said content.

9. The method of claim 3, further comprising the step of filtering said filtered offers in accordance with user profile information.

10. The method of claim 9, further comprising the step of updating said user profile information based on user activity.

11. The method of claim 1, further comprising the step of displaying said content and said offers on a processing device associated with said user.

12. The method of claim 1, further comprising the steps of contextually analyzing and indexing said offers, eliminating duplicate offers, and storing said indexed offers in said database.

13. System for contextually matching offers, comprising:

a database containing offers;

a browser for selecting a content by a user; and

a server for contextually analyzing said content to determine the context of said content; and

wherein said server is operable to select offers from said database matching the context of said content.

14. The system of claim 13, wherein said server is operable to filter said offers in accordance with a business rule to provide filtered offers.

15. The system of claim 14, wherein said server is operable to rank said filtered offers based on relevance to said content.

16. The system of claim 13, wherein said offers in said database being arranged into clusters, each cluster having pre-determined feature vectors and comprising contextually related offers; wherein said server is operable to generate a feature vector from said content, to compare said feature vector to said pre-determined feature vectors to determine contextually related clusters, and to select offers from contextually related clusters.

17. The system of claim 16, wherein said each cluster further comprising a center vector; wherein said server is operable to compare said feature vector to center vectors to determine at least one contextually related cluster and to select offers from said contextually related cluster.

18. The system of claim 17, wherein said center vector of said contextually related cluster being closest to said feature vector.

19. The system of claim 13, wherein each offer being at least one of the following: product, service, content item, advertisement and promotion.

20. The system of claim 13, wherein said server is operable to extract entities from said content.

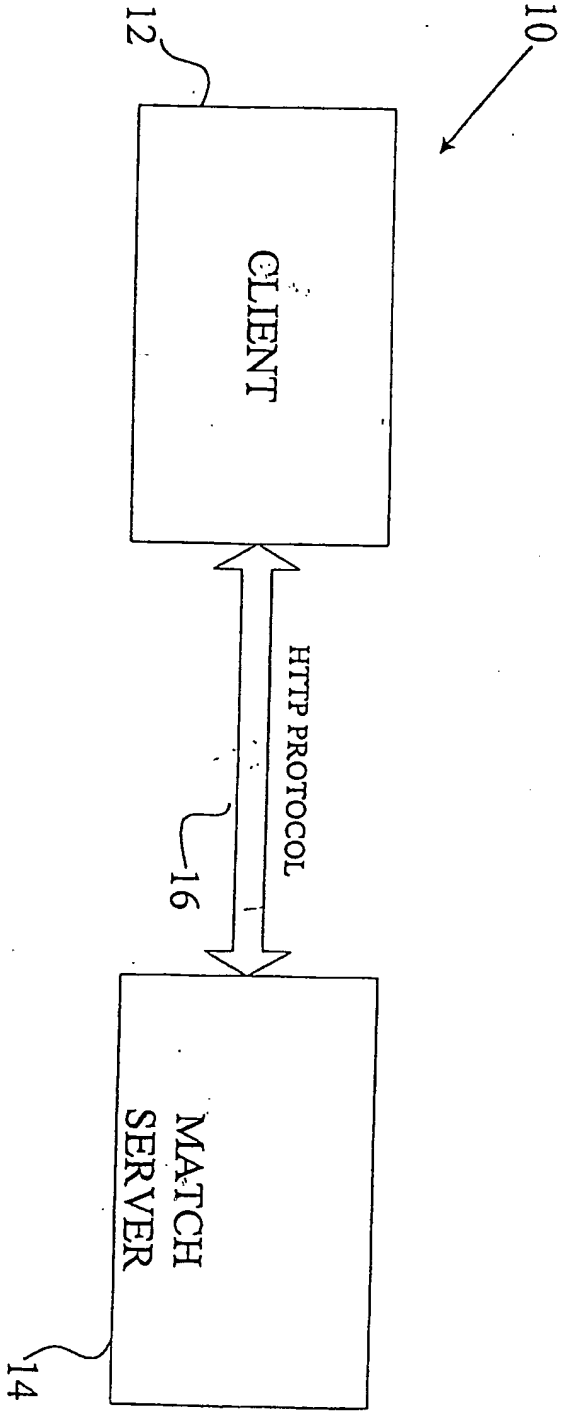
21. The system of claim 15, wherein said server is operable to filter said filtered offers in accordance with user profile information.

22. The system of claim 21, wherein said server is operable to update said user profile information based on user activity.

23. The system of claim 13, wherein said browser is operable to display said content and said offers on a processing device associated with said user.

24. The system of claim 13, wherein said server is operable to contextually analyze and index said offers, eliminate duplicate offers, and store said indexed offers in said database.

FIG. 1



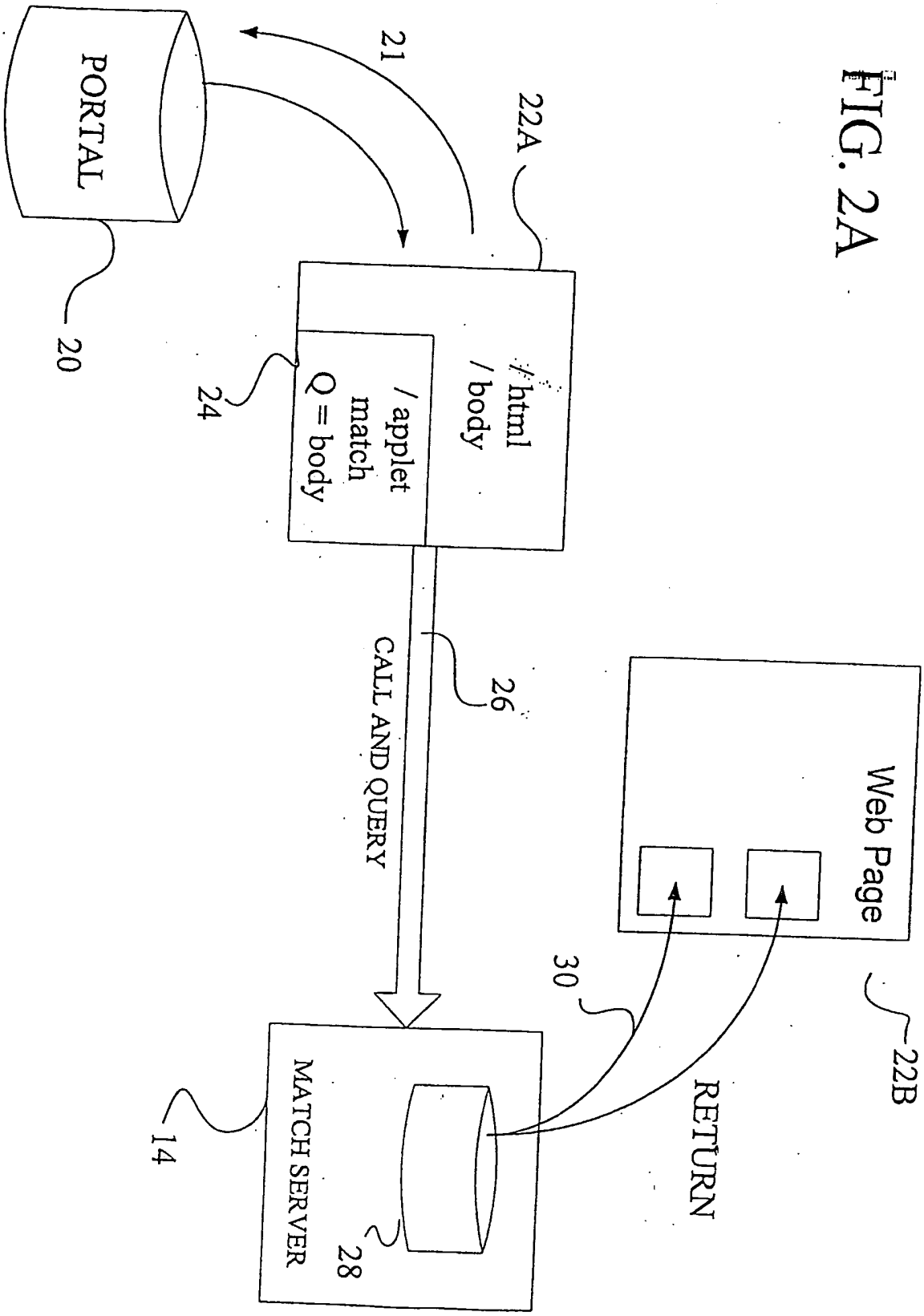

PORTAL
20
21
22A
/ html
/ body
/ applet
match
Q = body
24
CALL AND QUERY
26
Web Page
22B
30
RETURN
MATCH SERVER
14
28

FIG. 2A

FIG. 2B

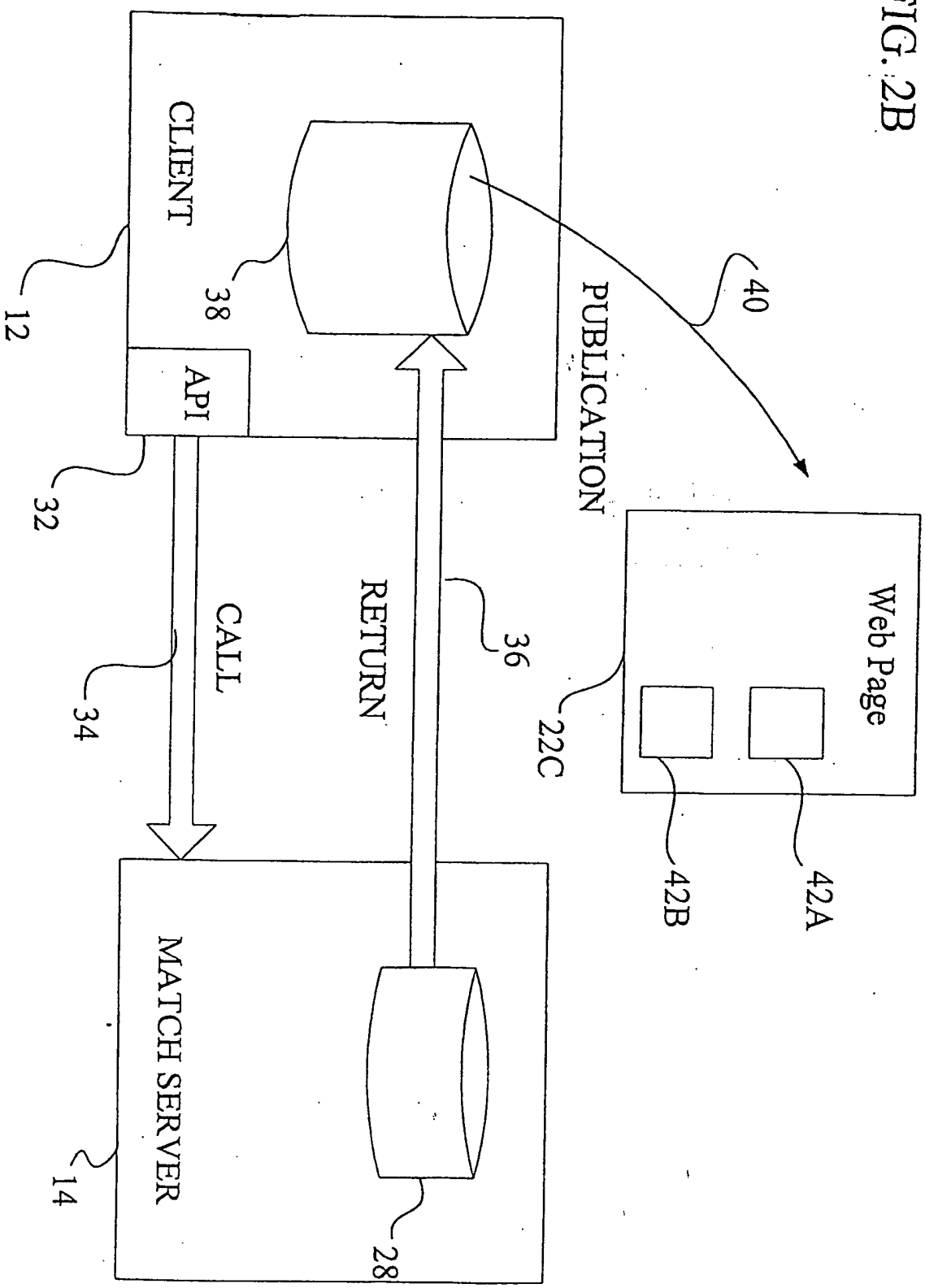

CLIENT
12
38
API
32
34
CALL
RETURN
36
PUBLICATION
40
Web Page
22C
42A
42B
MATCH SERVER
14
28

FIG. 3

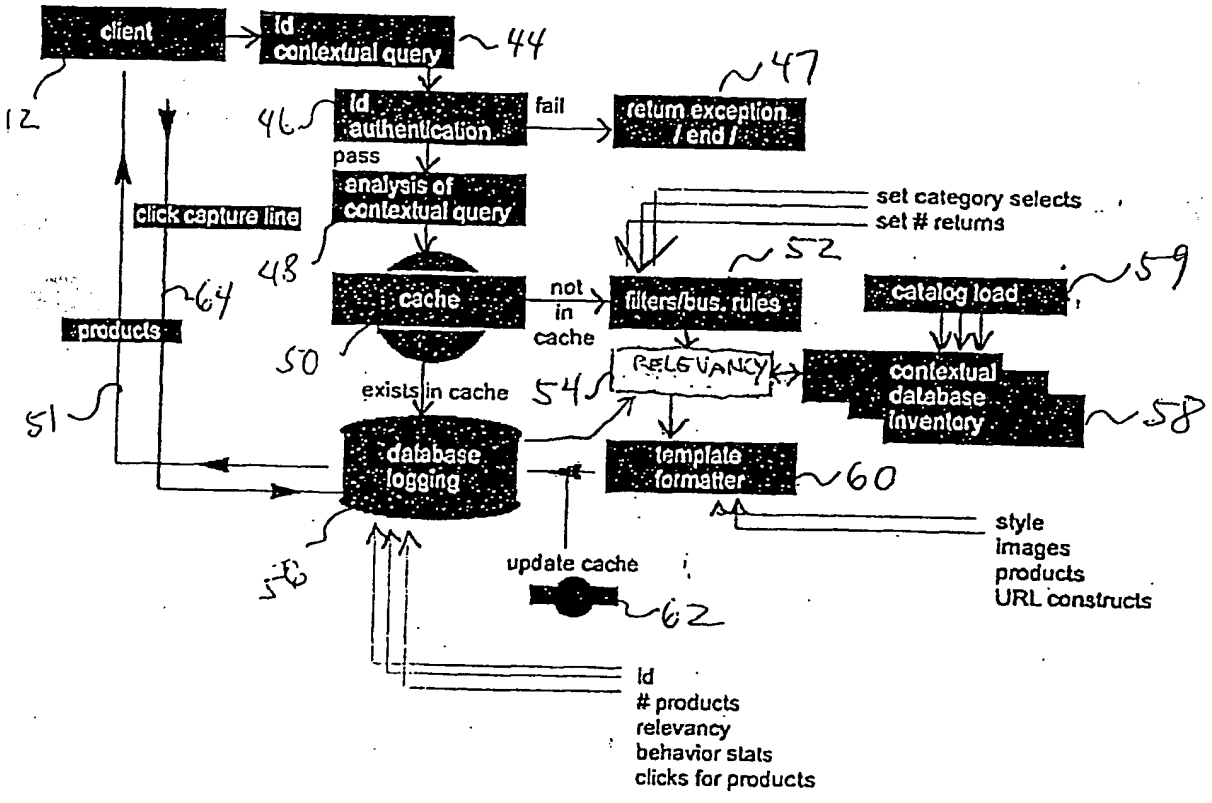

client
12
id contextual query
44
id authentication
46
fail
return exception / end /
47
pass
analysis of contextual query
48
click capture line
set category selects
set # returns
52
cache
50
not in cache
filters/bus. rules
catalog load
59
products
64
exists in cache
51
RELEVANCY
54
contextual database inventory
58
database logging
56
template formatter
60
style
images
products
URL constructs
update cache
62
id
products
relevancy
behavior stats
clicks for products

Fig. 4

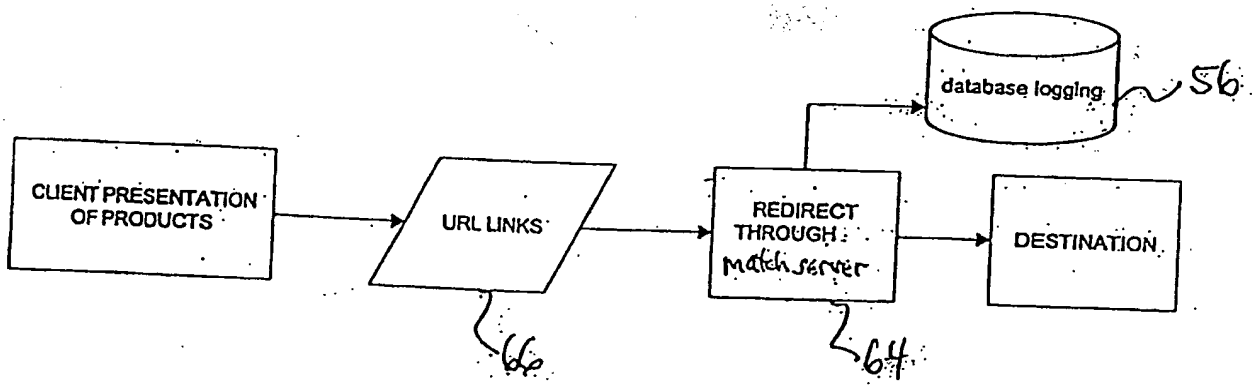

database logging
56
CLIENT PRESENTATION OF PRODUCTS
URL LINKS
REDIRECT THROUGH match server
DESTINATION
66
64

Fig. 5

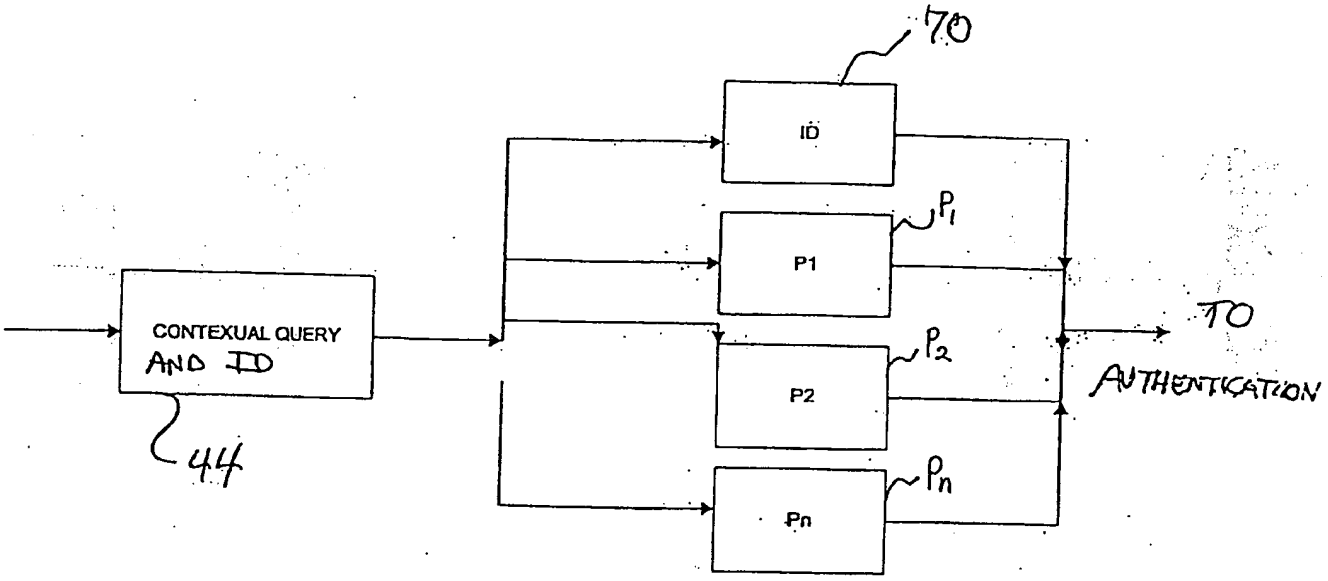

70
ID
P1
P1
P2
P2
Pn
Pn
CONTEXUAL QUERY
AND ID
44
TO
AUTHENTICATION

Fig. 6

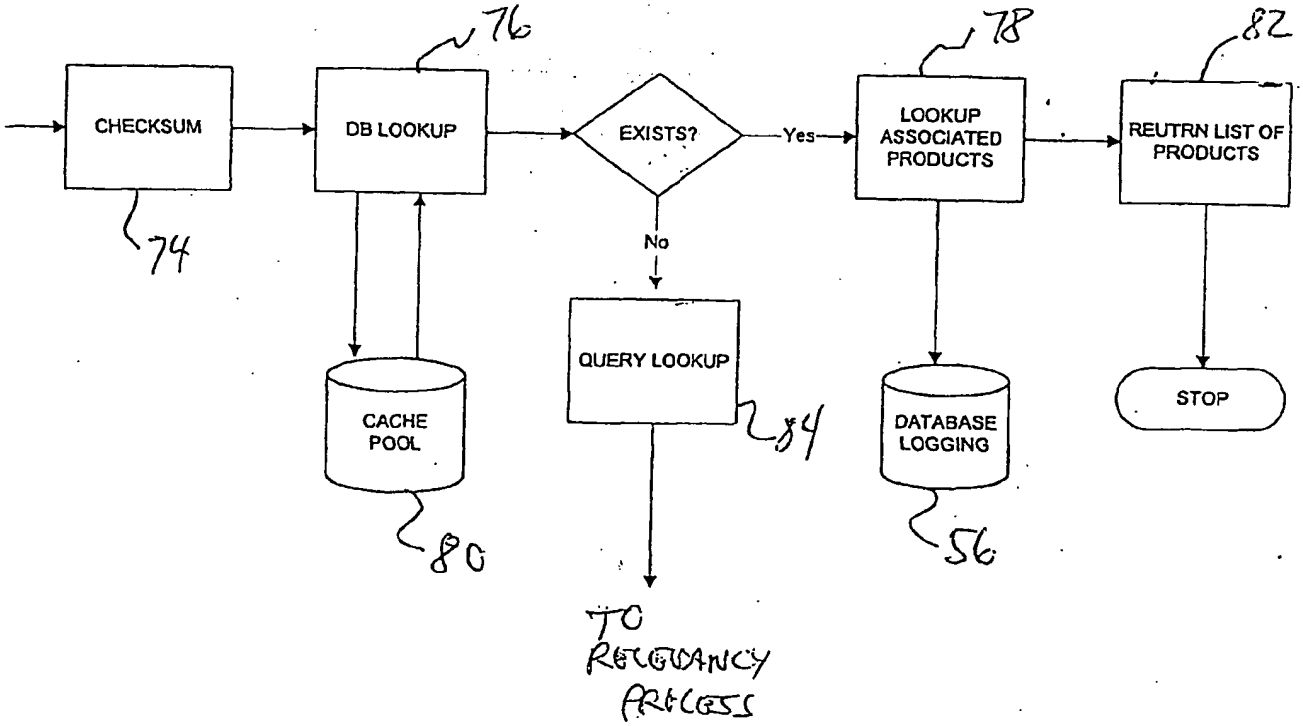

CHECKSUM
74
DB LOOKUP
76
CACHE POOL
80
EXISTS?
Yes
No
QUERY LOOKUP
84
TO RELEVANCY PROCESS
LOOKUP ASSOCIATED PRODUCTS
78
DATABASE LOGGING
56
REUTRN LIST OF PRODUCTS
82
STOP

Fig. 7A

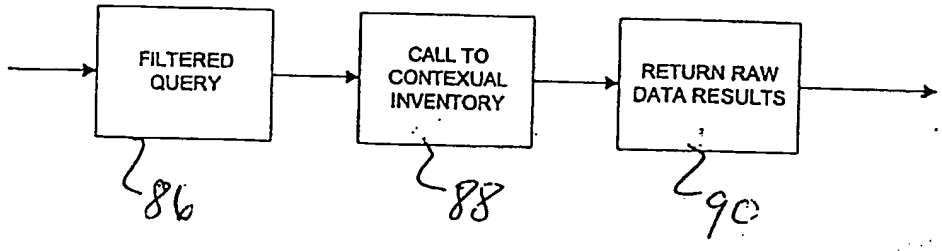

FILTERED QUERY
CALL TO CONTEXUAL INVENTORY
RETURN RAW DATA RESULTS
86
88
90

Fig. 7B

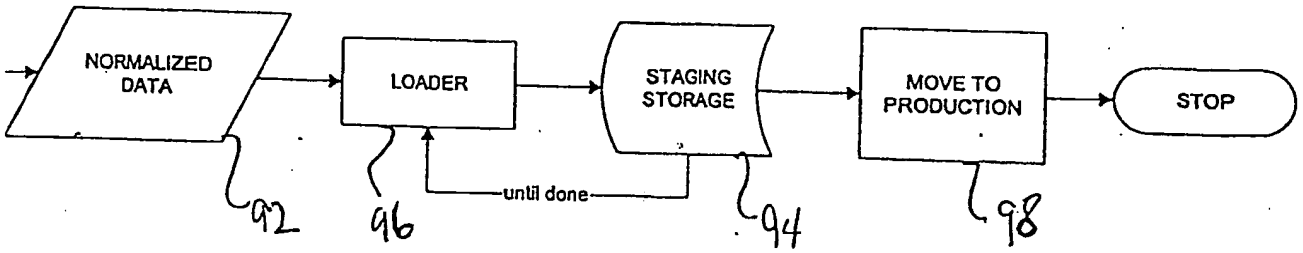

NORMALIZED DATA
LOADER
STAGING STORAGE
MOVE TO PRODUCTION
STOP
until done
92
96
94
98

Fig. 8

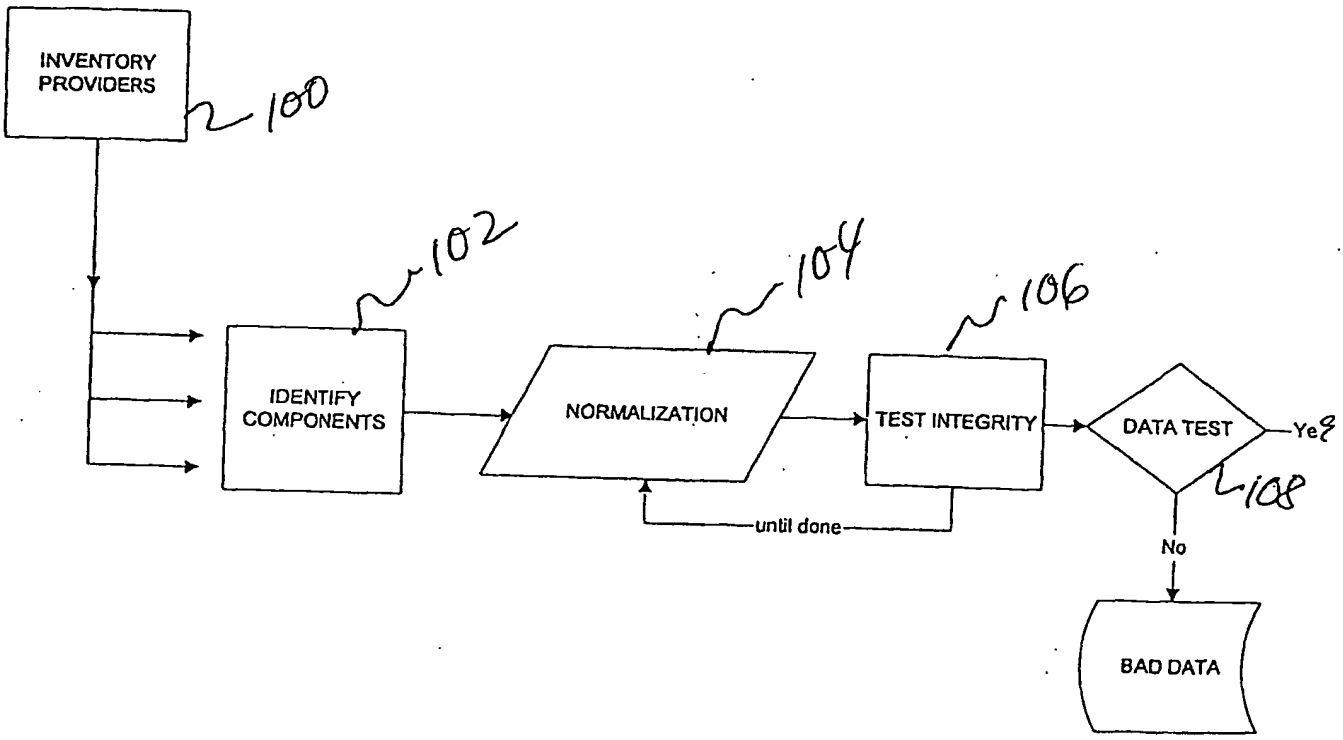

INVENTORY PROVIDERS
100
IDENTIFY COMPONENTS
102
NORMALIZATION
104
TEST INTEGRITY
106
DATA TEST
Yes
108
No
until done
BAD DATA

Fig. 8

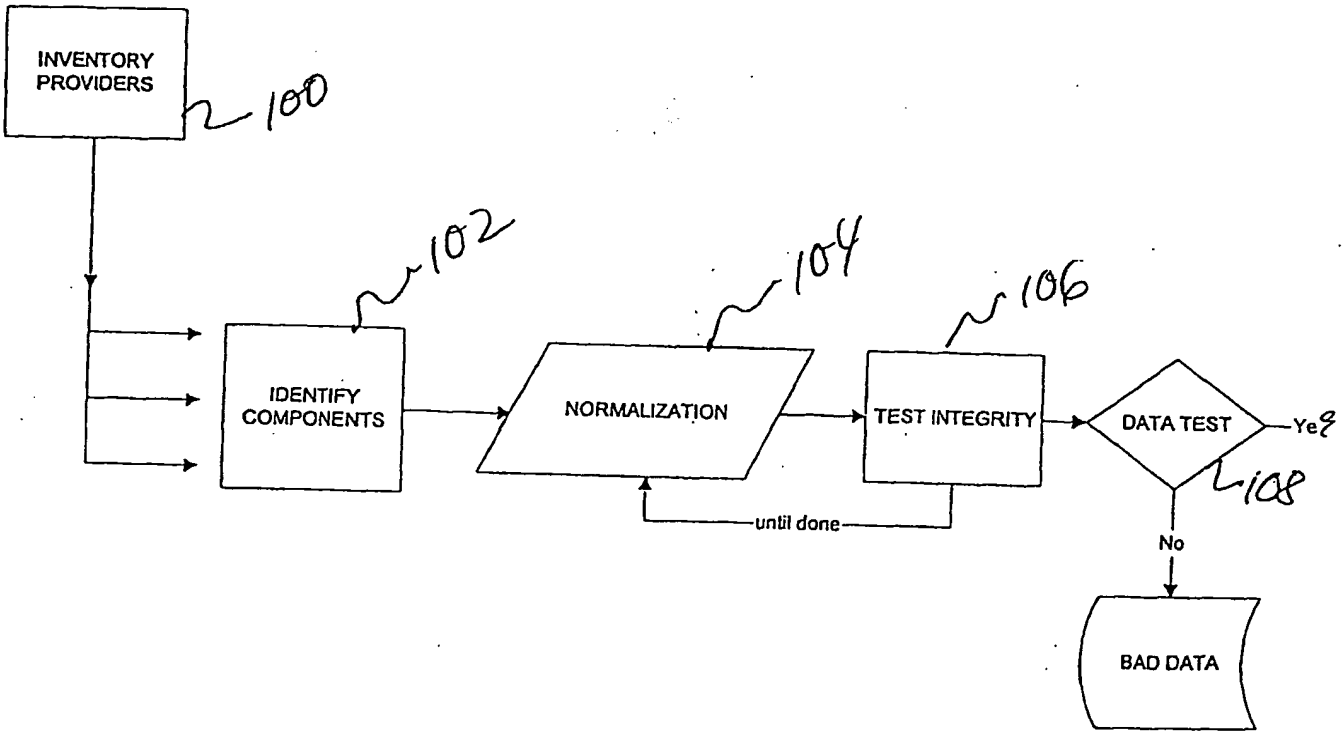

INVENTORY PROVIDERS
100
IDENTIFY COMPONENTS
102
NORMALIZATION
104
TEST INTEGRITY
106
DATA TEST
Yes
108
No
until done
BAD DATA

Fig. 10

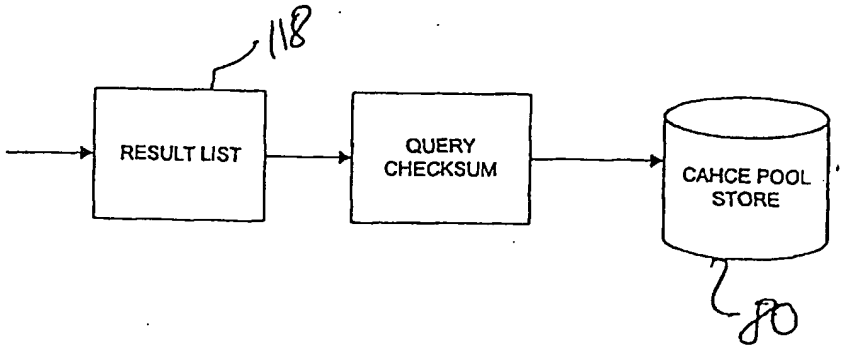


Fig. 11

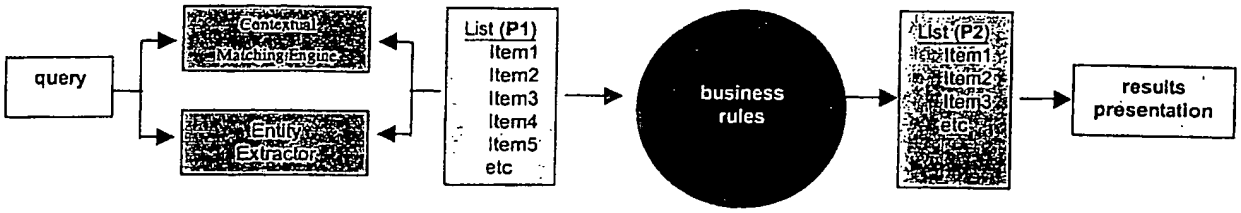

query
Contextual Matching Engine
Entity Extractor
List (P1)
Item1
Item2
Item3
Item4
Item5
etc
business rules
List (P2)
Item1
Item2
Item3
etc
results presentation

Fig. 12

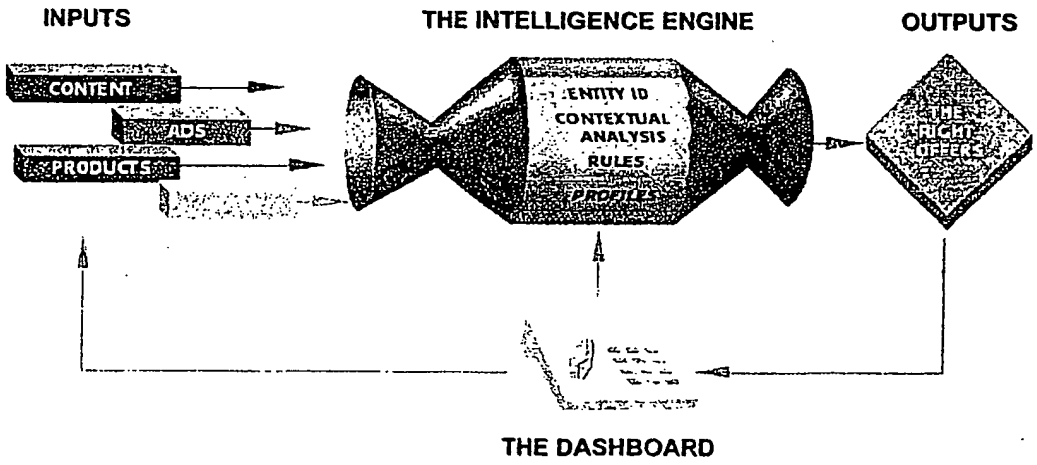

INPUTS
THE INTELLIGENCE ENGINE
OUTPUTS
CONTENT
ADS
PRODUCTS
ENTITY ID
CONTEXTUAL ANALYSIS
RULES
PROFILES
THE RIGHT OFFER
THE DASHBOARD

Fig. 13

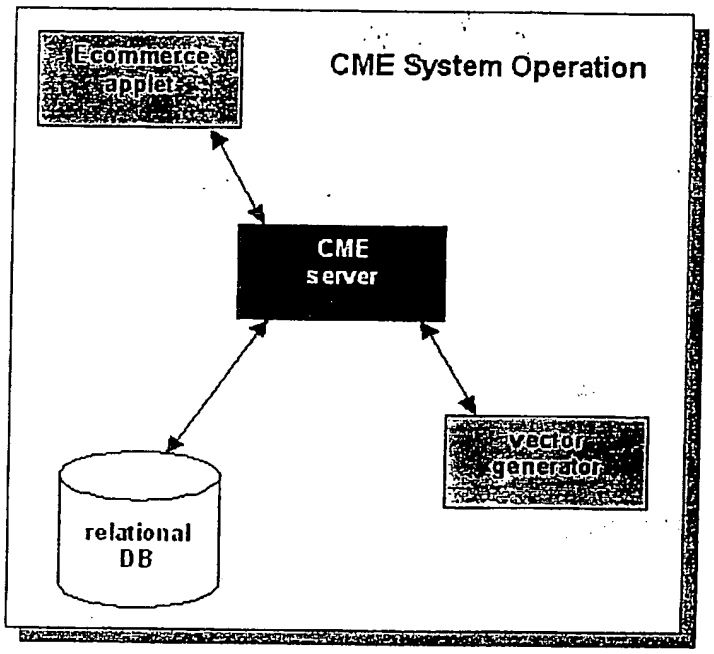

CME System Operation
E commerce applet
CME server
vector generator
relational DB

Fig. 14

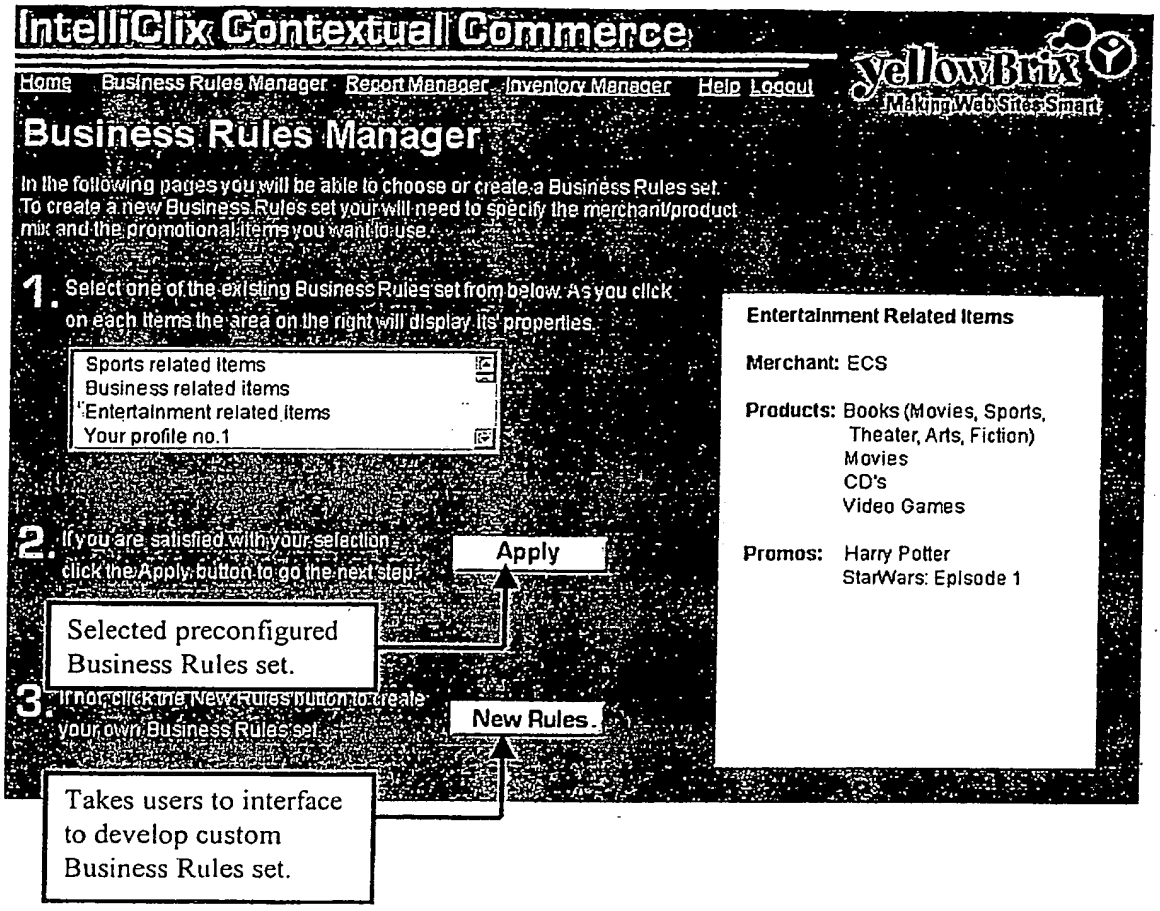

IntelliClix Contextual Commerce
Home Business Rules Manager Report Manager Inventory Manager Help Logout
YellowBrix
Making Web Sites Smart
Business Rules Manager
In the following pages you will be able to choose or create a Business Rules set. To create a new Business Rules set your will need to specify the merchant/product mix and the promotional items you want to use.
1. Select one of the existing Business Rules set from below. As you click on each items the area on the right will display its properties.
Sports related items
Business related items
Entertainment related items
Your profile no.1
Entertainment Related Items
Merchant: ECS
Products: Books (Movies, Sports, Theater, Arts, Fiction)
Movies
CD's
Video Games
Promos: Harry Potter
StarWars: Episode 1
2. If you are satisfied with your selection click the Apply button to go the next step.
Apply
Selected preconfigured Business Rules set.
3. If not, click the New Rules button to create your own Business Rules set.
New Rules.
Takes users to interface to develop custom Business Rules set.

Fig. 15

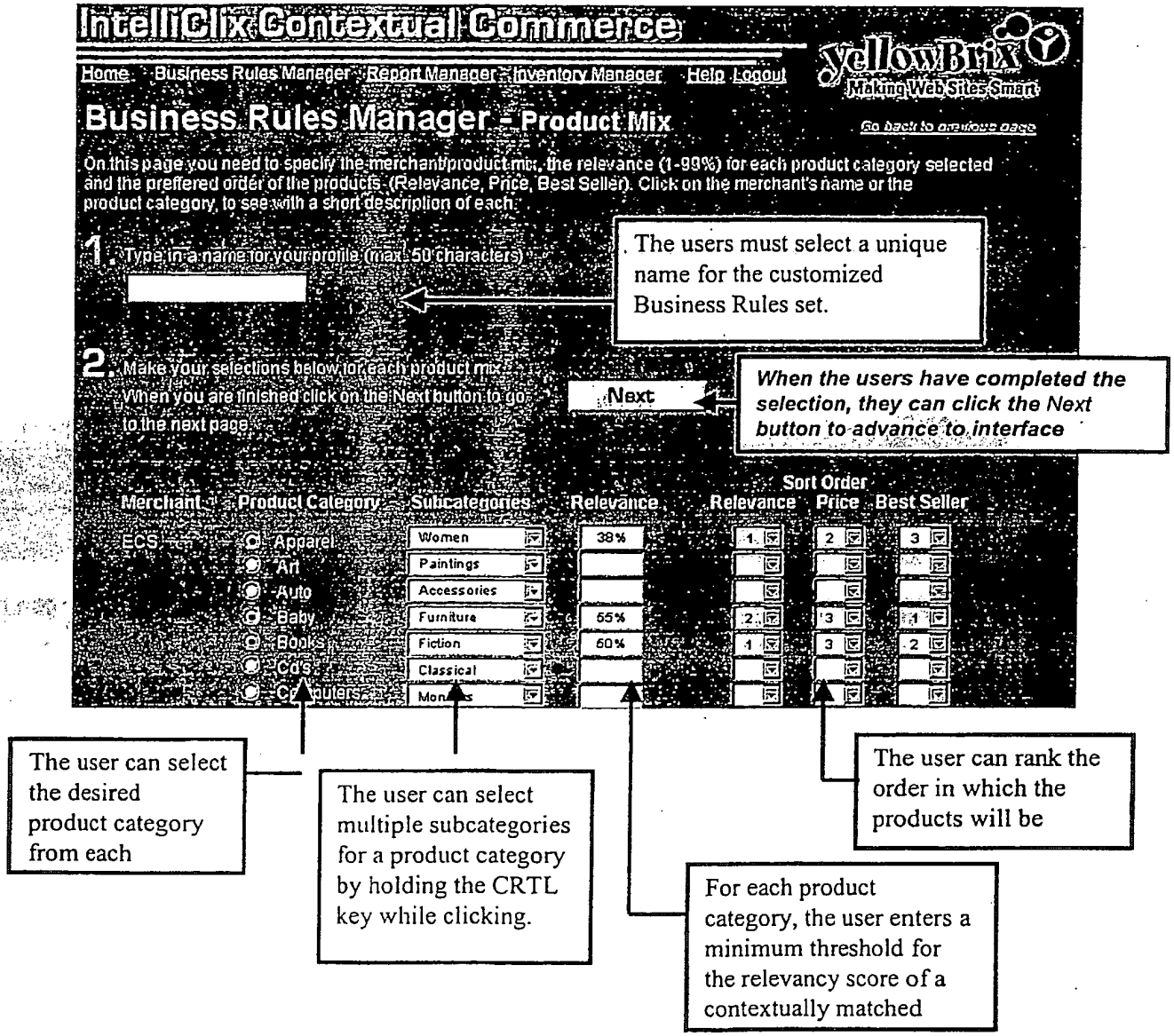

IntelliClix Contextual Commerce
YellowBrix
Making Web Sites Smart
Home Business Rules Manager Report Manager Inventory Manager Help Logout
Business Rules Manager - Product Mix
Go back to previous page
On this page you need to specify the merchant/product mix, the relevance (1-99%) for each product category selected and the preffered order of the products. (Relevance, Price, Best Seller). Click on the merchant's name or the product category, to see with a short description of each.
1. Type in a name for your profile (max. 50 characters)
The users must select a unique name for the customized Business Rules set.
2. Make your selections below for each product mix. When you are finished click on the Next button to go to the next page.
Next
When the users have completed the selection, they can click the Next button to advance to interface
Sort Order
Merchant
Product Category
Subcategories
Relevance
Relevance
Price
Best Seller
ECS
Apparel
Art
Auto
Baby
Books
Cds
Computers
Women
Paintings
Accessories
Furniture
Fiction
Classical
38%
55%
60%
1
2
3
2
3
1
1
3
2
The user can select the desired product category from each
The user can select multiple subcategories for a product category by holding the CRTL key while clicking.
For each product category, the user enters a minimum threshold for the relevancy score of a contextually matched
The user can rank the order in which the products will be

Fig. 16

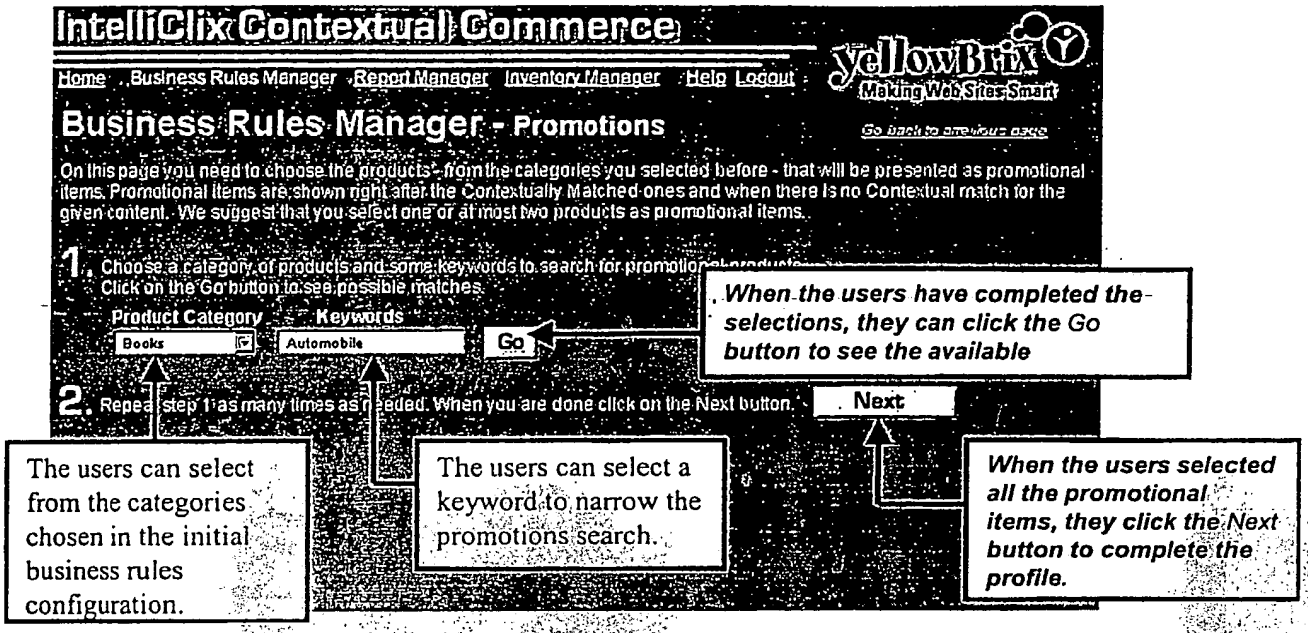
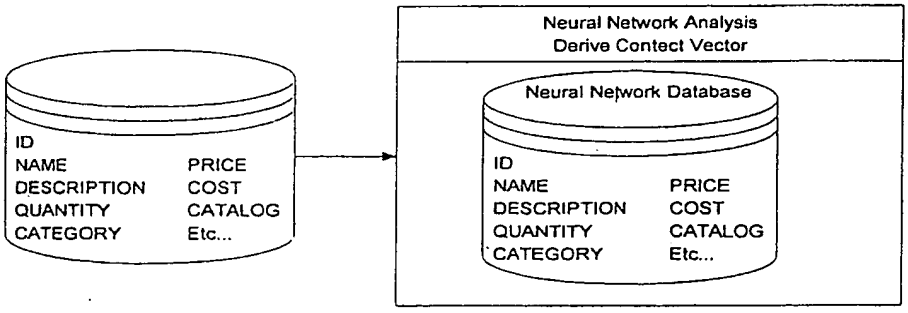

IntelliClix Contextual Commerce
Home
Business Rules Manager
Report Manager
Inventory Manager
Help
Logout
YellowBrix
Making Web Sites Smart
Business Rules Manager - Promotions
Go back to previous page
On this page you need to choose the products - from the categories you selected before - that will be presented as promotional items. Promotional items are shown right after the Contextually Matched ones and when there is no Contextual match for the given content. We suggest that you select one or at most two products as promotional items.
1. Choose a category of products and some keywords to search for promotional products
Click on the Go button to see possible matches
Product Category
Keywords
Books
Automobile
Go
When the users have completed the selections, they can click the Go button to see the available
2. Repeat step 1 as many times as needed. When you are done click on the Next button.
Next
The users can select from the categories chosen in the initial business rules configuration.
The users can select a keyword to narrow the promotions search.
When the users selected all the promotional items, they click the Next button to complete the profile.

Fig. 17


ID
NAME
PRICE
DESCRIPTION
COST
QUANTITY
CATALOG
CATEGORY
Etc...
Neural Network Analysis
Derive Contect Vector
Neural Network Database
ID
NAME
PRICE
DESCRIPTION
COST
QUANTITY
CATALOG
CATEGORY
Etc...

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/41713

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) :G06F 17/30
US CL :707/1, 2, 3, 5, 10, 201, 501; 709/ 219; 705/14.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 707/1, 2, 3, 5, 10, 201, 501; 709/ 219; 705/14.

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Please See Extra Sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,819,291 A (HAIMOWITZ et al) 06 October 1998, col. 1, lines 11-47 and lines 54-67, col. 2, lines 1-29 and lines 55-67, col. 3, lines 1-67, col. 4, lines 1-20 and lines 60-67, col. 5, lines 1-49, col. 6, lines 1-67, col. 7, lines 1-4 and lines 29-65, col. 8, lines 1-67, col. 9, lines 1-67, and col. 10, lines 1-31.	1-24
Y	US 5,948,061 A (MERRIMAN et al) 07 September 1999, col. 2, lines 7-40, col. 4, lines 21-64, col. 6, lines 3-59, col. 7, lines 2-14 and lines 32-43, col. 10, lines 37-57, and col. 14, lines 28-62.	1-24

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 06 MARCH 2001	Date of mailing of the international search report 16 APR 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer ELLA COLBERT James R. Matthews Telephone No. (703) 308-7064

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US00/41713

B. FIELDS SEARCHED
Electronic data bases consulted (Name of data base and where practicable terms used):

WEST
Search terms: database matching, vectors, targeted marketing, products, advertising, analyzing, filter, index, user profile, ranking, internet, world wide web, network, web.

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☑ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.